

# 영상을 적용한 인공지능을 이용한 **Robot Arm Placing** 기술 개발

Baek Young Jin

2020.07.15

# INDEX

01. 연구 목표

02. 광학 흐름(Optical Flow)

03. 극좌표 변환(Polar Coordinate)

04. 광학 흐름과 극좌표 변환 알고리즘 결합

05. 결론 및 향후 연구방향

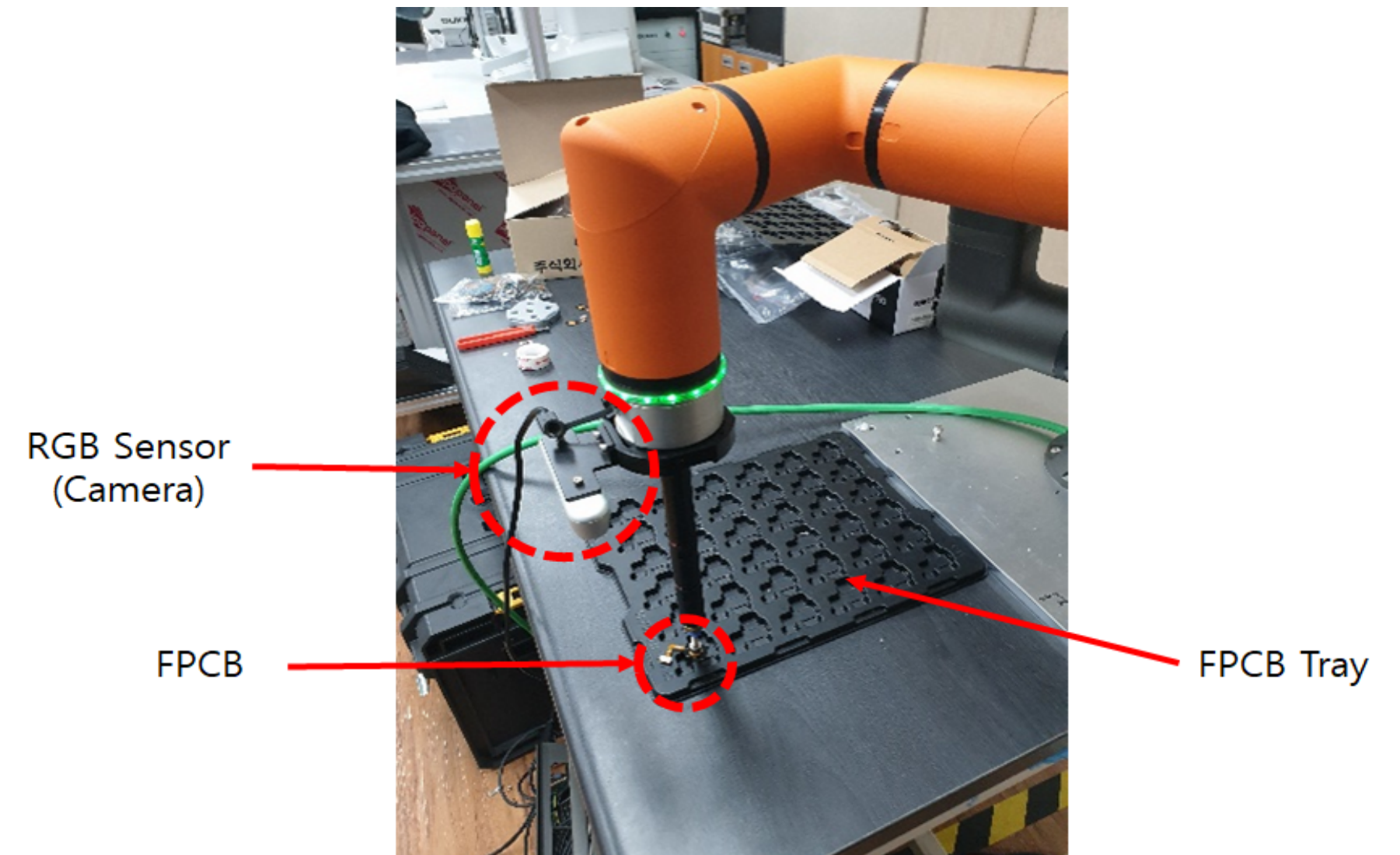
# 01.연구 목표

## 기존 연구 분석

- 로봇 팔을 이용하여 물체를 Placing하는 기술은 여러 응용분야에서 연구되어 다양한 기술이 존재한다.
- 현재는 물체에 따라 사람이 로봇 팔을 직접 튜닝하여 사용하고 있다.
- 강화학습을 사용하는 알고리즘이 많이 개발 중이나 사용되는 리소스가 크다.

## 연구 목표

- RGB센서(카메라) 만으로 트레이에 FPCB를 삽입하는 알고리즘을 개발한다.
- 최소한의 리소스로 최대한 빠른 시간, 높은 정확도를 보장한다.  
(사람이 작업할 때와 비교해 크게 오차가 발생하면 안된다.)



<Robot Arm Image>

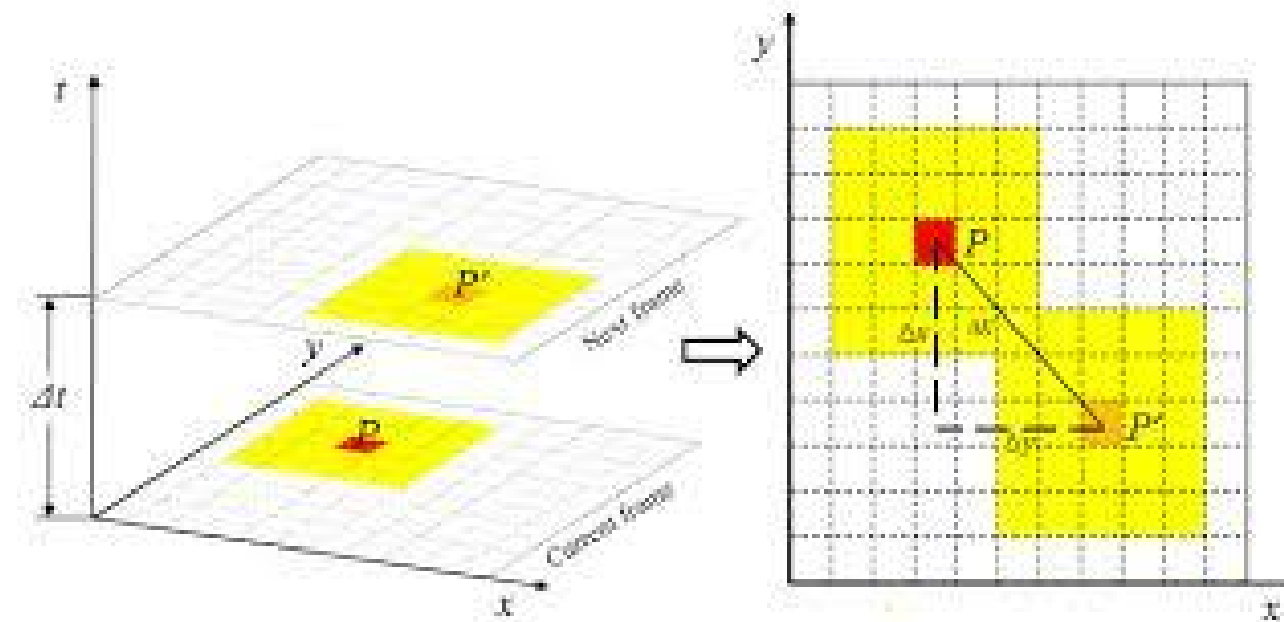


<RGB Sensor에서 획득한 Image>

## 02.Optical Flow Definition

### Definition

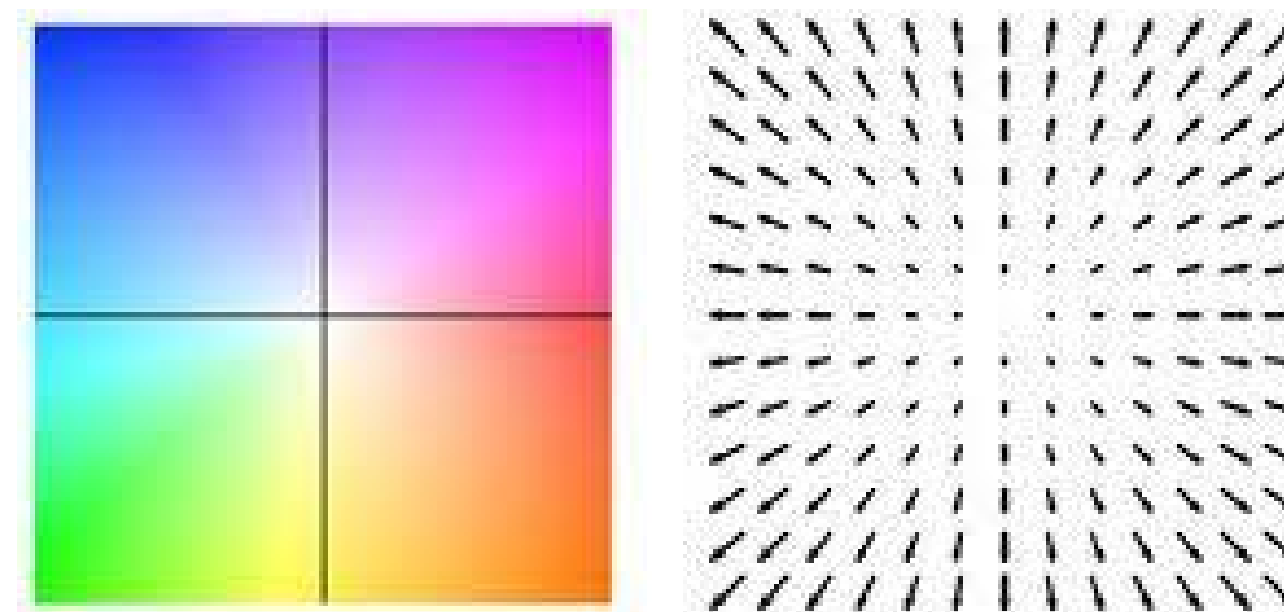
연속한 두 frame 사이에서 각 Pixel 의 Motion 을  
나타내는 Vector Map



### Expression

Vector : 모든 Pixel에서 표현이 불가능하다.  
한눈에 보기 힘들며 난잡하다.

HSV : 색상, 채도, 명도로 Optical Flow를 표현



## 02.Optical Flow FlowNet

*Past*

HardWare의 한계

DeepLearning의 부재

Ground Truth 추출의 어려움  
( DataSet의 부족 )

*Present*

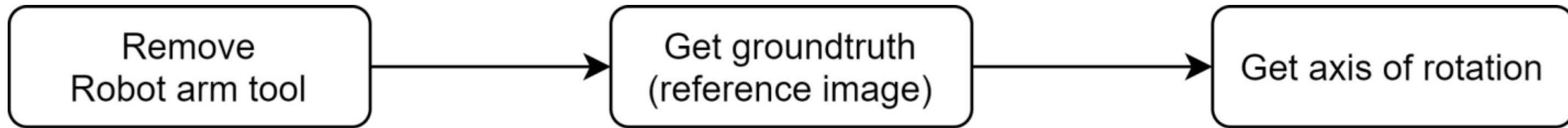
GPU의 발달  
GPU 병렬처리

CNN의 등장

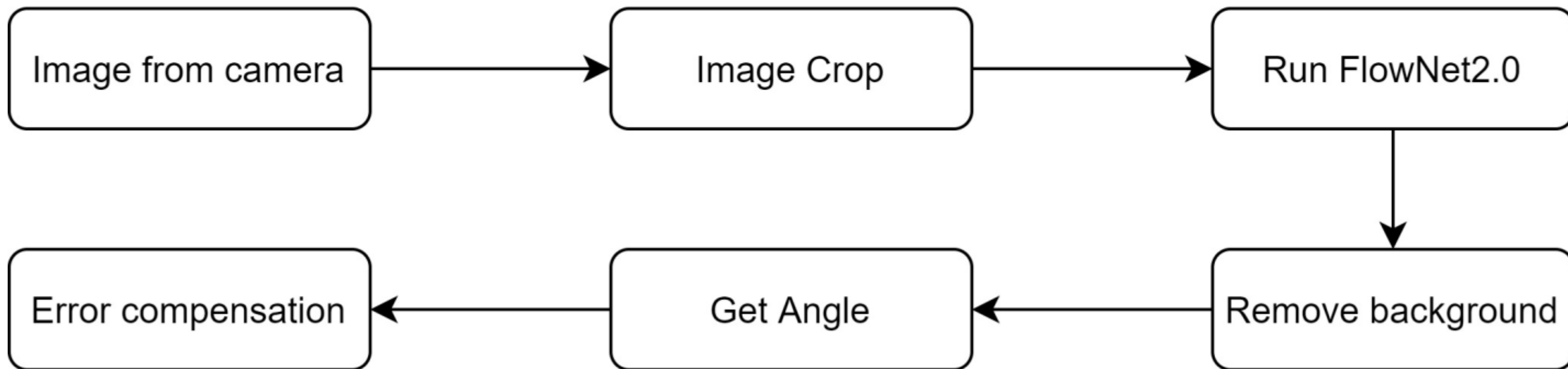
Test 위한 Flying Chairs 제공  
학습 후 Sintel 등으로 추가 진행

실제 문제에 적용할만한 속도 및 성능을 낼 수 있음

## 02.Optical Flow Flow Chart using FlowNet2.0



< Preprocessing >

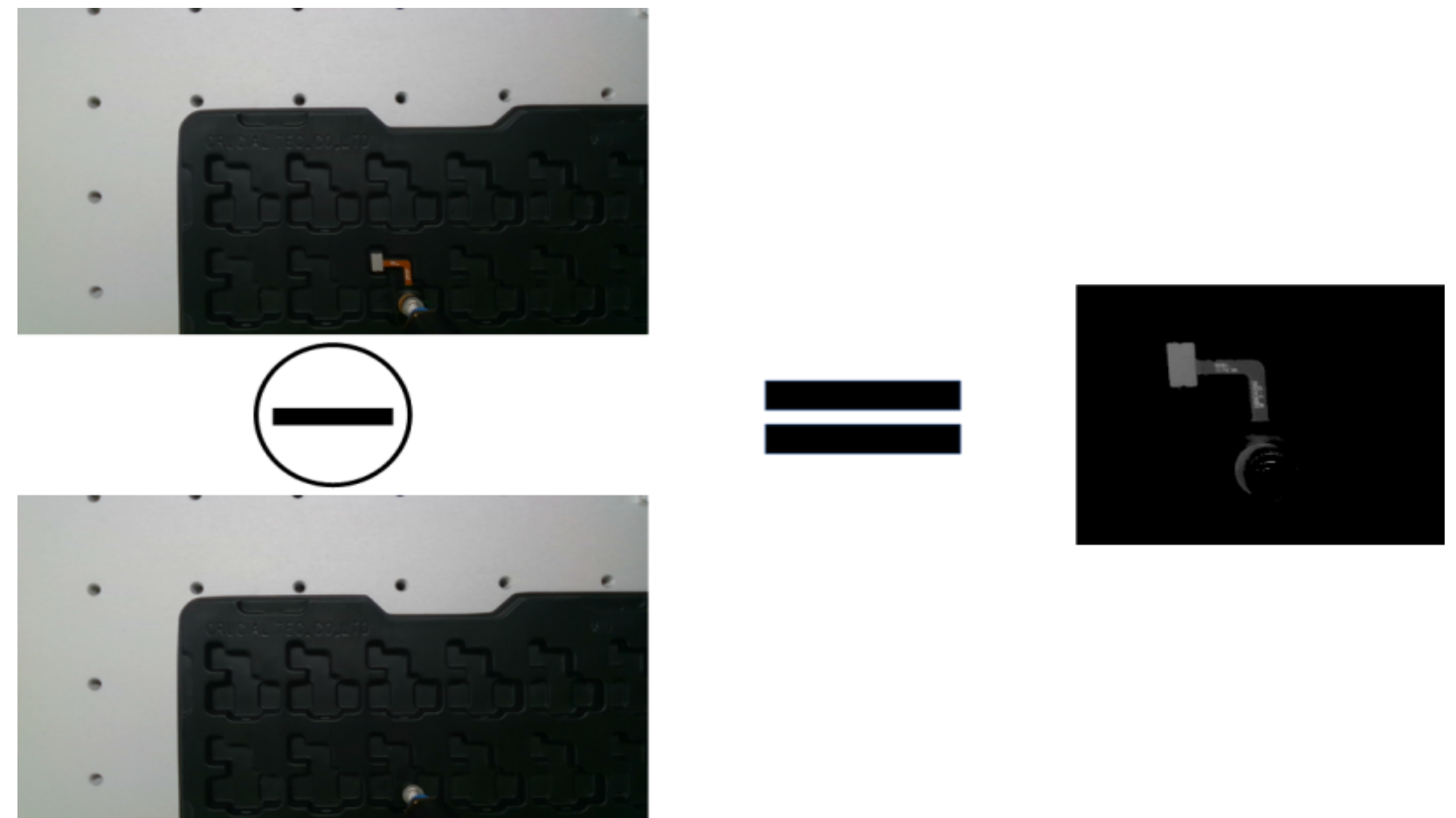


< Processing using FlowNet2.0 >

## 02.Optical Flow 2.1. 로봇 팔 Tool 제거 2.2. 정답 이미지 획득



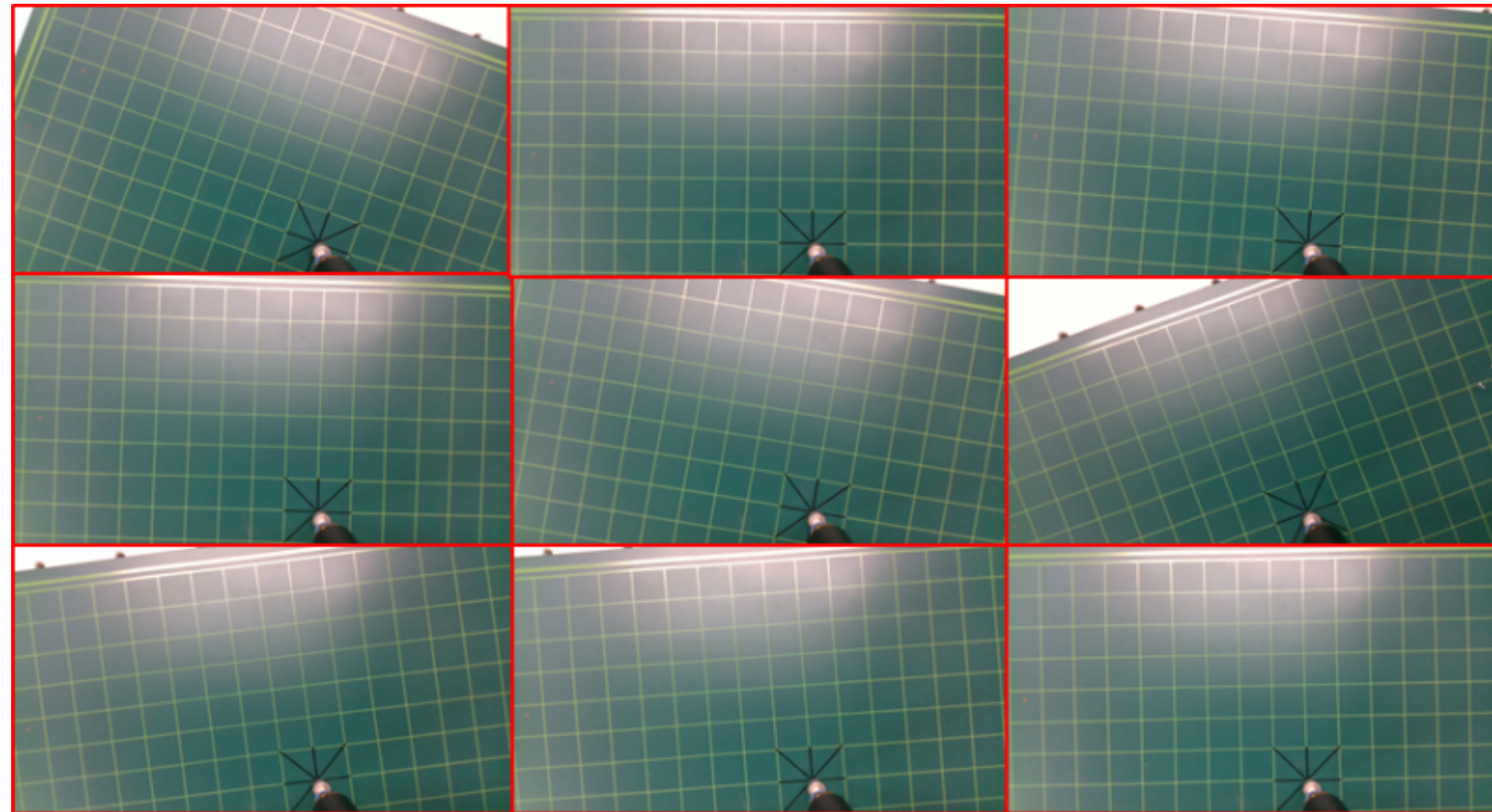
<Robot Arm Tool 제거 후 이미지>



<Groundtruth Image(reference image) 구하는 과정>



## 02.Optical Flow 2.3. Robot Arm 회전축 좌표 획득 2.4. 이미지 Crop 및 FlowNet 실행



각도(°)	좌표	SSIM 점수
-19	(783,635)	0.912389
-6	(782,635)	0.945469
-3	(782,636)	0.947600
3	(784,639)	0.947236
6	(782,634)	0.946610
11	(782,635)	0.943099
21	(785,635)	0.925145

<SSIM 알고리즘을 이용해 Robot Arm의 회전축 좌표 획득>



<From 'rotation Image' to 'reference image' 결과>



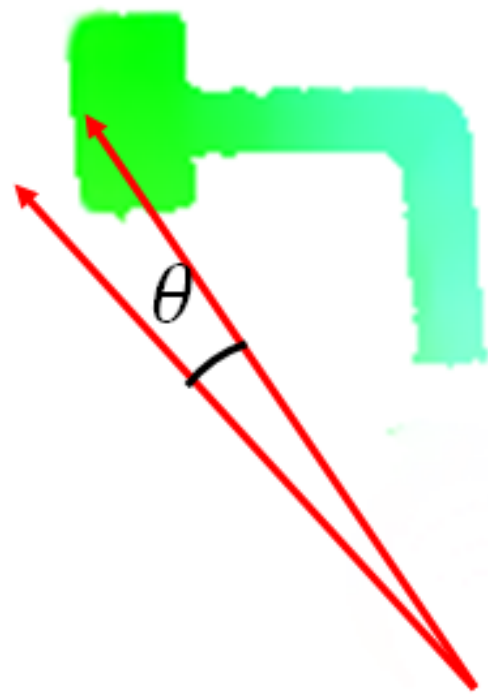
<reference image를 이용해 배경 제거>



<From 'reference Image' to 'rotation image' 결과>

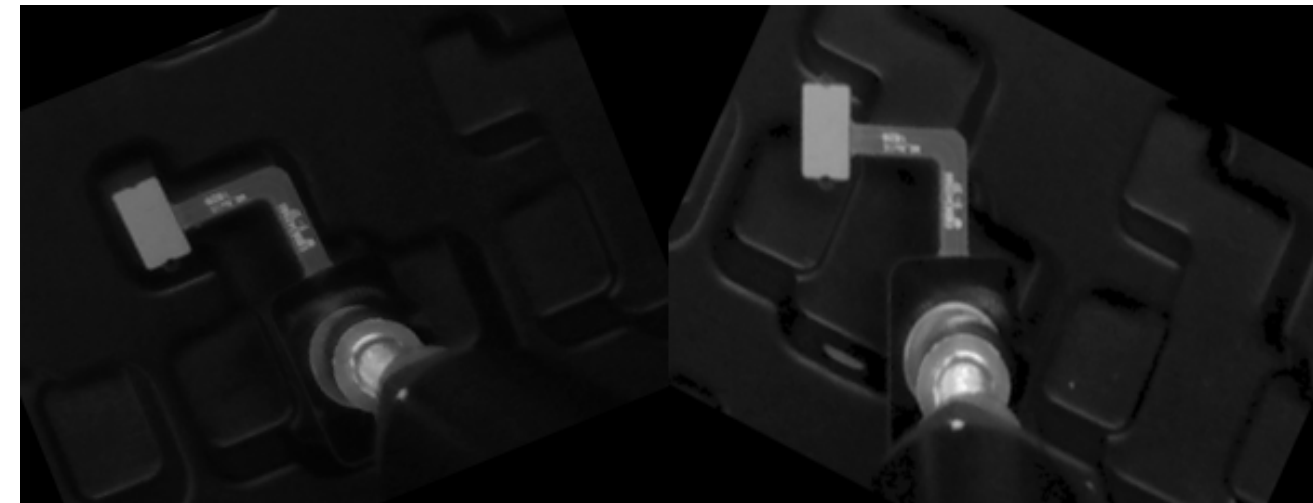


## 02.Optical Flow 2.5. 회전각 획득 2.6. 오차 보정

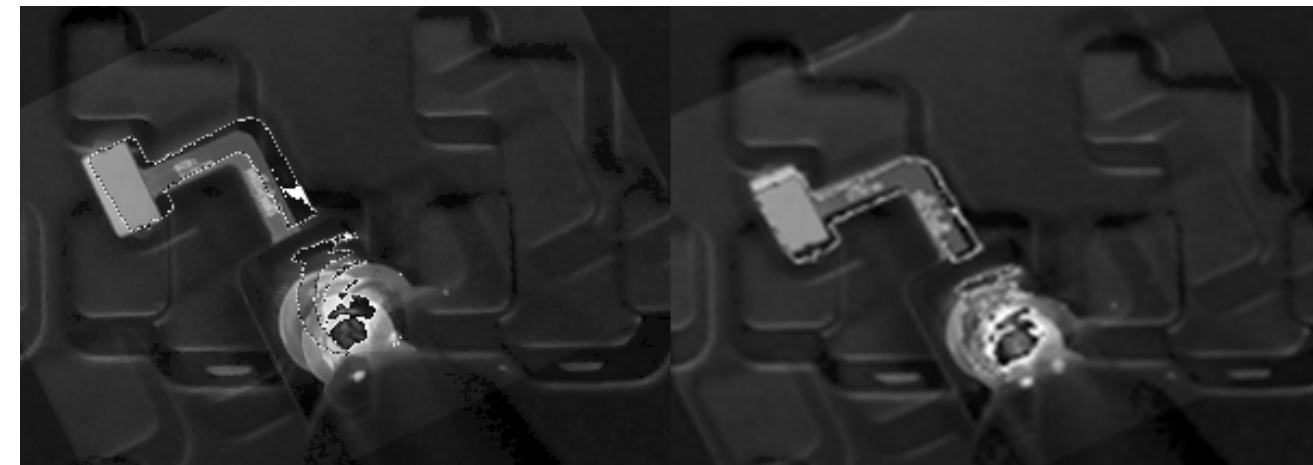


$$\theta = \cos^{-1}\left(\frac{v1 \cdot v2}{\|v1\| \times \|v2\|}\right)$$

<벡터의 내적을 이용해 회전각 획득>

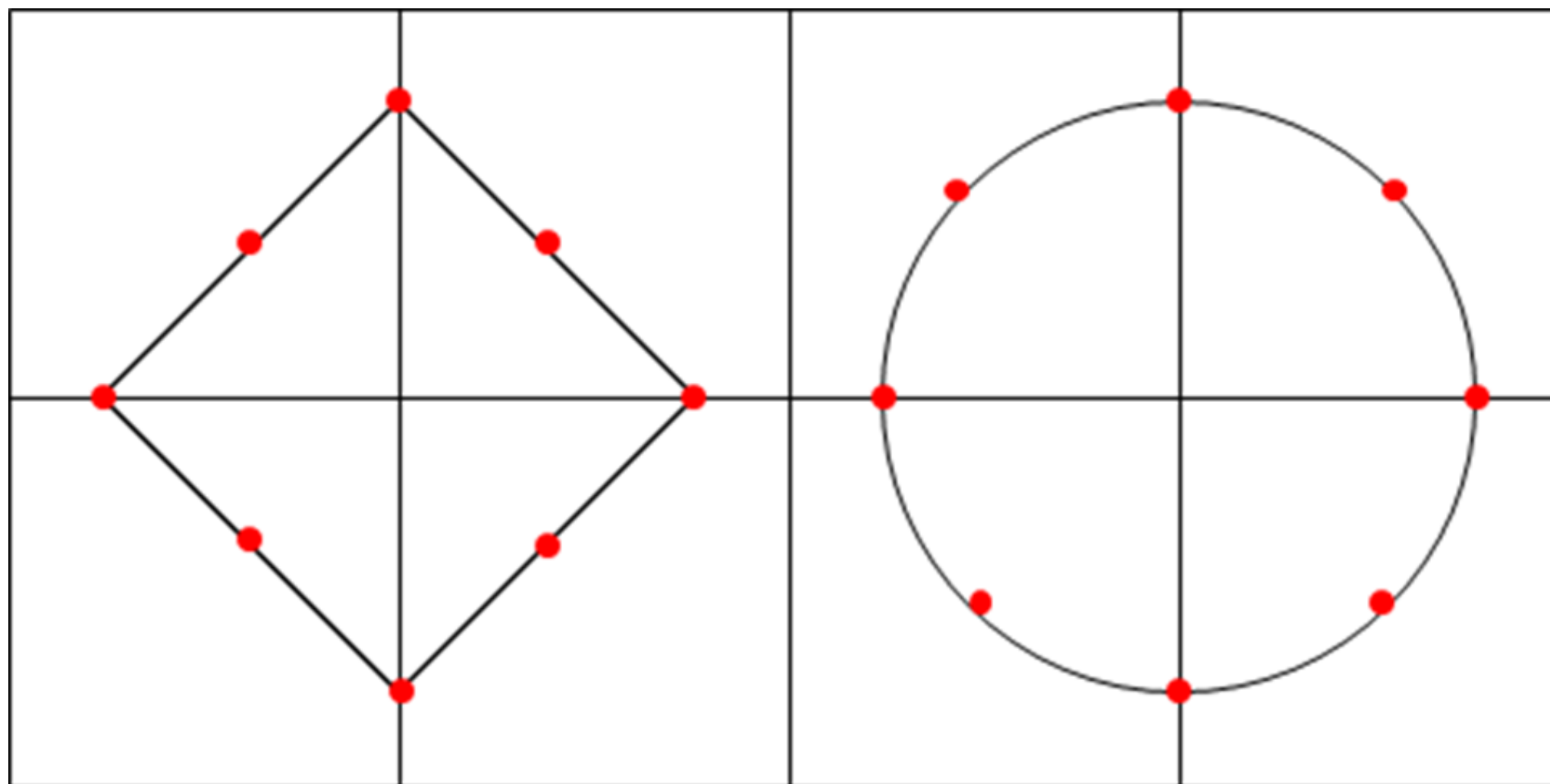


<획득한 회전각을 이용해 reference image, rotation image 회전 결과>



<FlowNet 한 번 실행한 결과 영상과 두 번 실행한 결과 영상 비교>

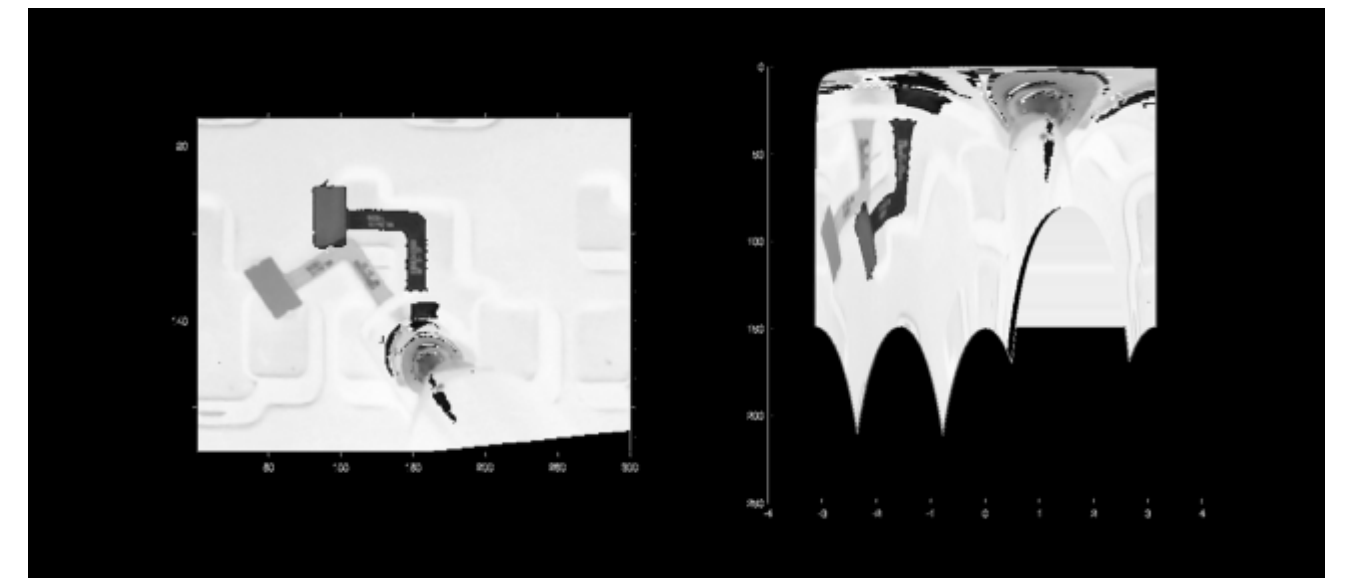
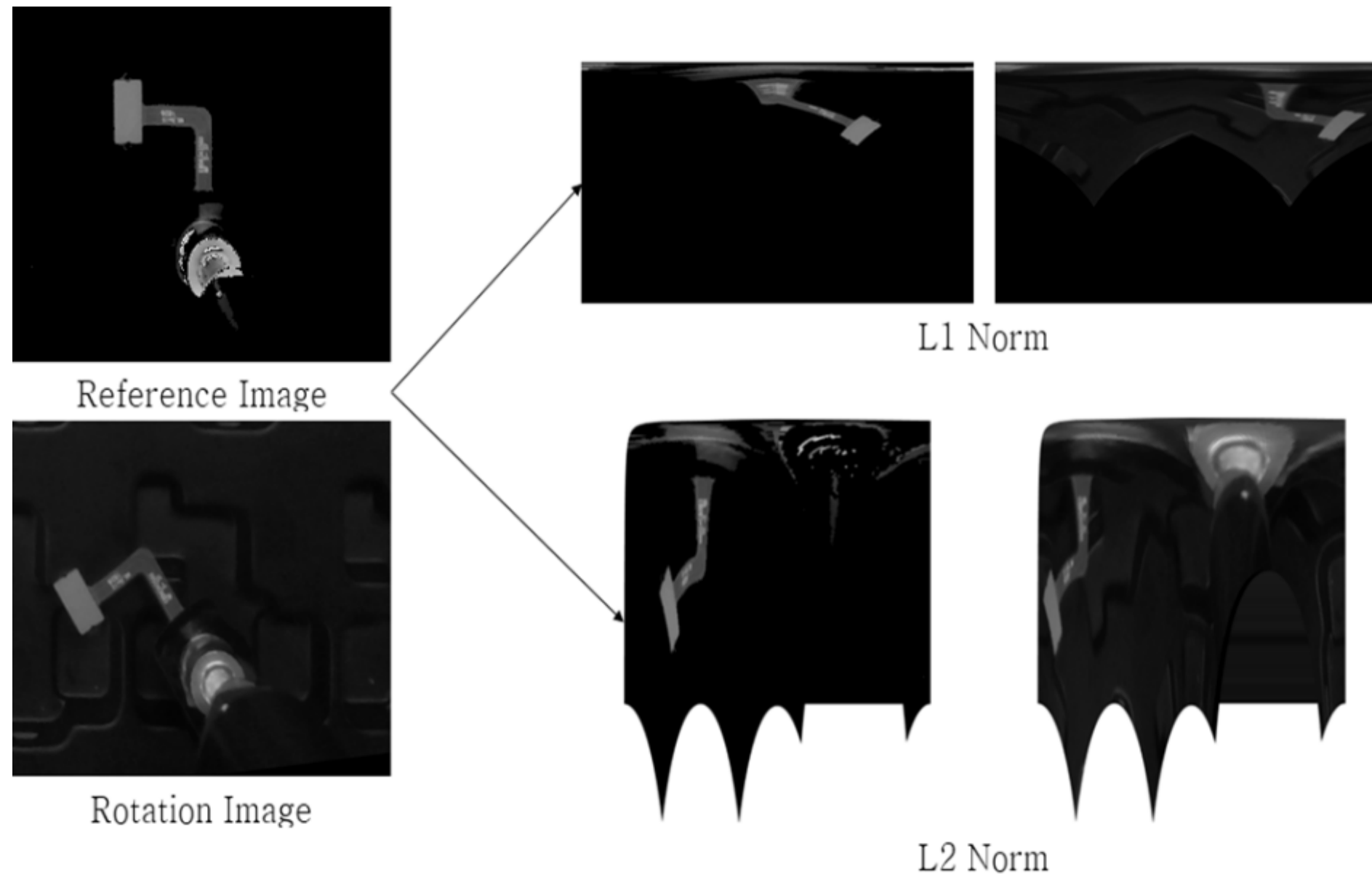
### 03. Polar Coordinate L1 Norm, L2 Norm



$$L_1 = \sum_i^n |x_i|$$
$$\|A\|_1 = 1$$
$$= |\alpha_0| + |\alpha_1|$$

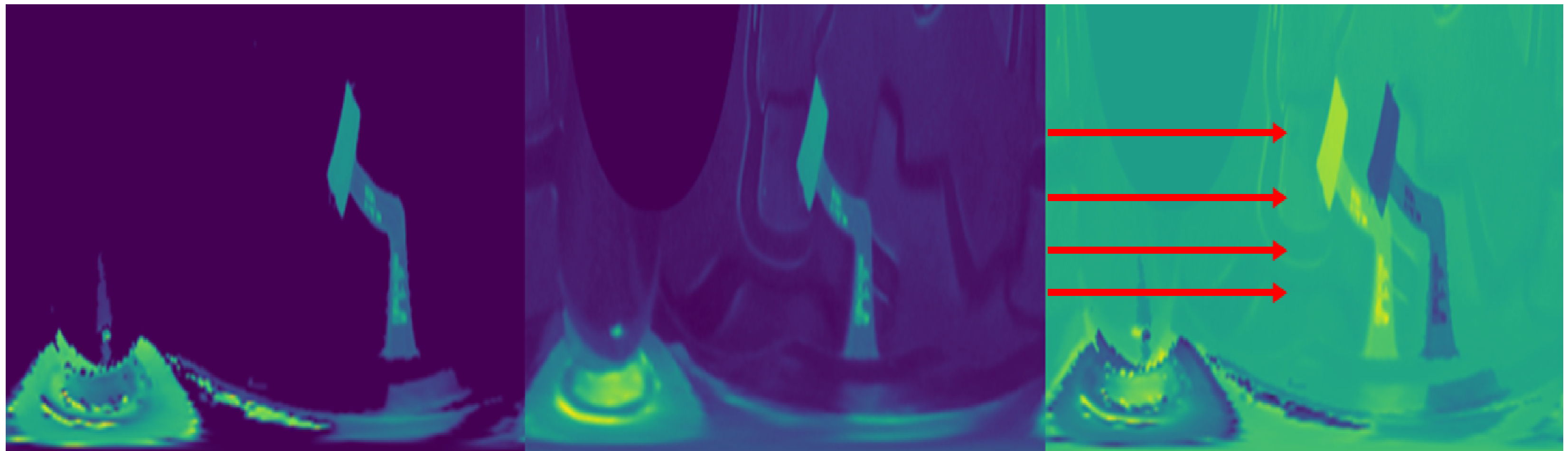
$$L_2 = \sqrt{\sum_i^n x_i^2}$$
$$\|A\|_2 = 1$$
$$= \sqrt{\alpha_0^2 + \alpha_1^2}$$

### 03.Polar Coordinate



<L2 norm 적용 후 reference image와 rotation image 비교 영상>

## 03.Polar Coordinate

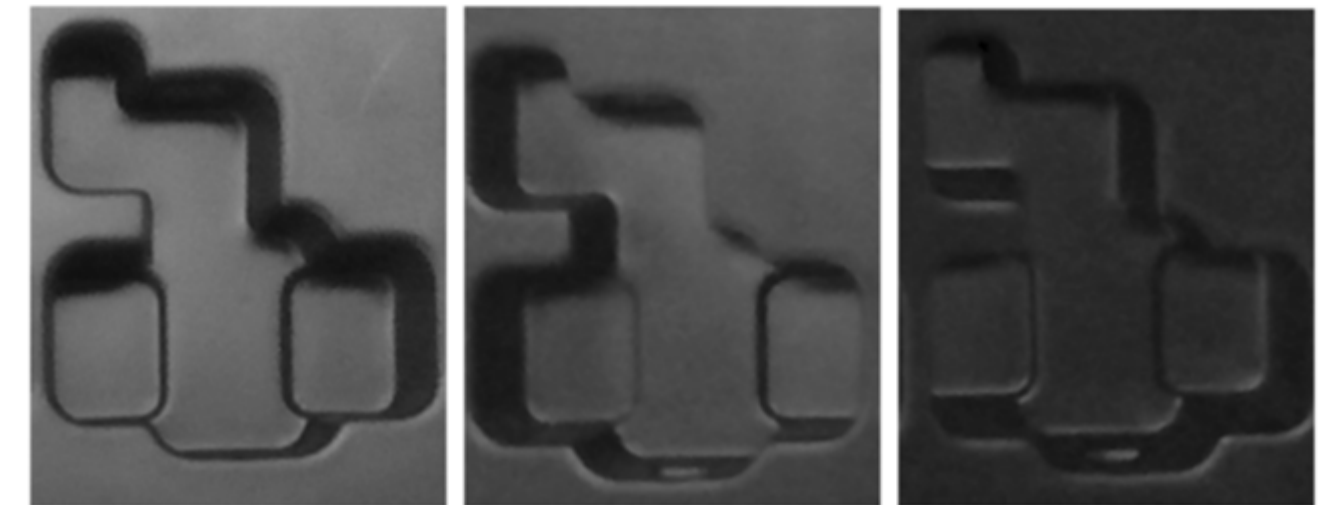


<1D correlation을 이용해 shift거리를 구하면 각도를 획득할 수 있다.>

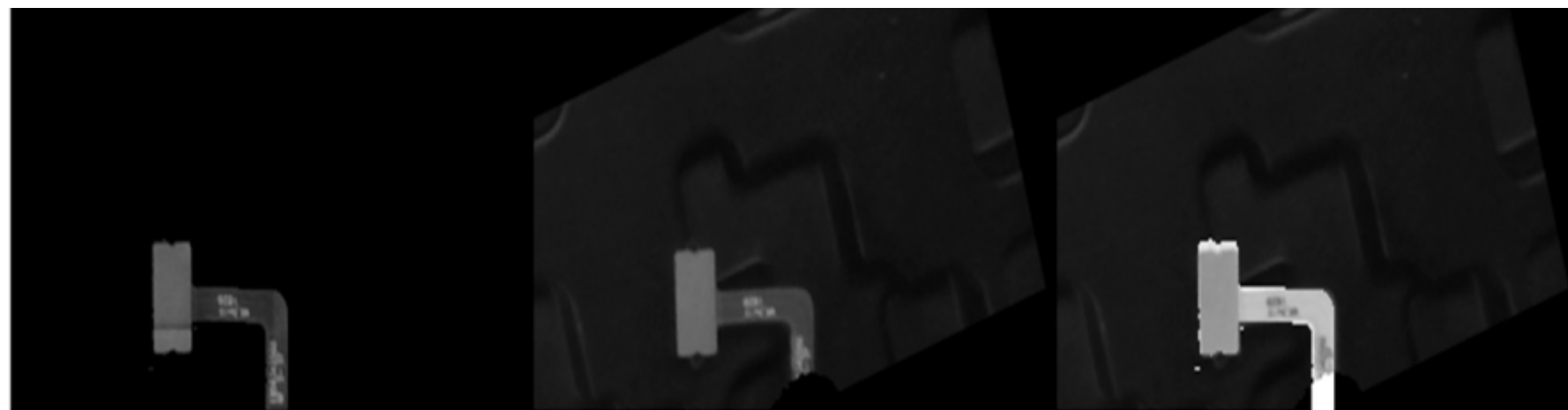
## 04. 광학 흐름과 극좌표 변환 알고리즘 결합



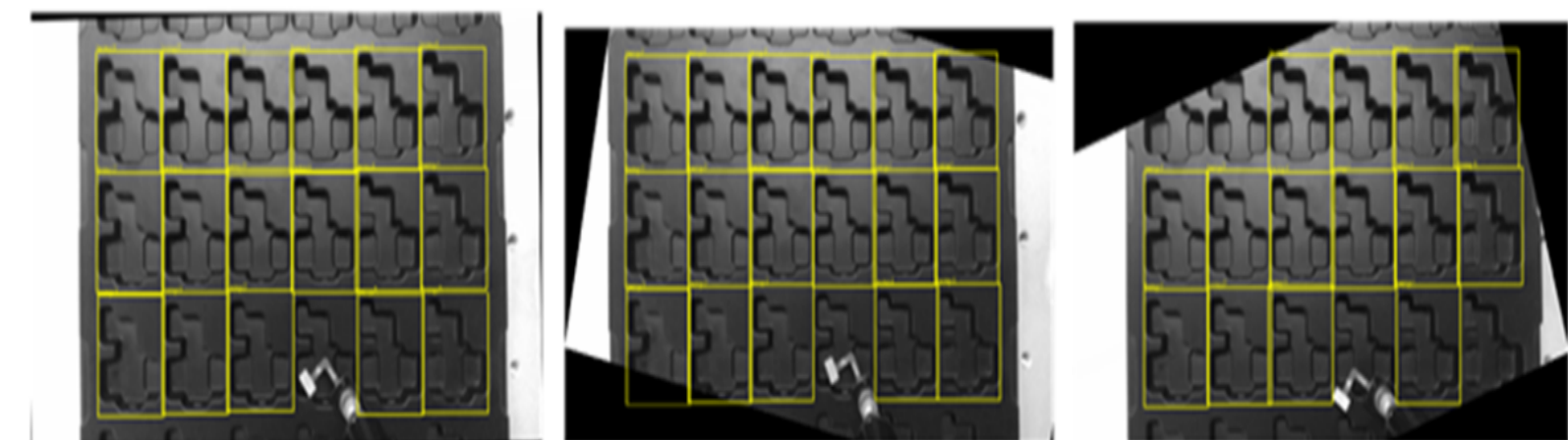
<Polar coordinate에 적합하게 180°만 남김 reference image와 잘못 잡은 케이스>



<좌측 상단, 우측 상단, 우측 하단의 Template>



<FlowNet을 통해 보정한 reference image와 결과로 얻은 회전 영상>



<MTM 적용 결과>



## 05. 결론 및 향후 연구방향

### 결론

-RGB Sensor(카메라)만을 이용해 성공적으로 FPCB의 회전각을 추출했다.

다른 물리적 센서가 개입하지 않았기 때문에 다양한 모양의 FPCB에 적용 가능한 알고리즘을 개발했다.

-시뮬레이션 환경이 아닌 실제 획득한 image들로 알고리즘을 개발해 실제 문제에 적용이 수월하다.

### 향후 연구방향

-잘못 잡은 케이스에 어느 정도 성공적으로 기능을 하지만 더 나빠질 경우 에러가 발생한다.

-변화가 큰 경우 FlowNet의 소요 시간을 보장할 수 없어 총 소요 시간을 보장할 수 없다.

# Thank you

Baek Young Jin

2020.07.15